REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)* 14-02-2017	**2. REPORT TYPE** Final Technical	**3. DATES COVERED** *(From - To)* 04/01/2015 - 08/31/2016

4. TITLE AND SUBTITLE An Investigation of Kernel Data Attacks and Countermeasures	**5a. CONTRACT NUMBER**
	5b. GRANT NUMBER N00014-15-1-2136
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) Haining Wang	**5d. PROJECT NUMBER**
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) University of Delaware, 210 Hullihen Hall, Newark, DE 19711	**8. PERFORMING ORGANIZATION REPORT NUMBER** ELEG33232815000
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Office of Naval Research, Linda Shipp - 703-696-8559, linda.shipp@navy.mil	**10. SPONSOR/MONITOR'S ACRONYM(S)** ONR
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for Public Release, distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Altering in-memory kernel data, attackers are able to manipulate the running behaviors of operating systems without injecting any malicious code. This type of attack is called kernel data attack. Intuitively, the security impact of such an attack seems minor, and thus, it has not yet drawn much attention from the security community. In this project, we thoroughly investigate kernel data attack, showing that its damage could be as serious as kernel rootkits. Especially, by tampering with kernel data, we demonstrate that attackers can stealthily subvert various kernel security mechanisms and develop a new keylogger, which is more stealthy than existing keyloggers.. By classifying kernel data into different categories and handling them separately, we propose a defense mechanism and evaluate its efficacy with real experiments. We expect the results of this project to enable transformative rethinking of the current kernel data security issues in a computer system.

15. SUBJECT TERMS

Kernel Data, Rootkit, Keylogger, Countermeasure

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Haining Wang
U	U	U	UU	3	**19b. TELEPHONE NUMBER** *(Include area code)* 757-813-4390

Final Technical Report for

"An Investigation of Kernel Data Attacks and Countermeasures"
Award No: N00014-15-1-2136

Haining Wang

Department of Electrical and Computer Engineering
University of Delaware
Newark, DE 19716
Email: hnw@udel.edu

1 Overall Technical Achievement

Altering in-memory kernel data, attackers are able to manipulate the running behaviors of operating systems without injecting any malicious code. This type of attack is called kernel data attack. Intuitively, the security impact of such an attack seems minor, and thus, it has not yet drawn much attention from the security community. In this project, we have thoroughly investigated kernel data attack, showing that its damage could be as serious as kernel rootkits, and then have proposed effective countermeasures. More specifically, by tampering with kernel data, we have first demonstrated that attackers can stealthily subvert various kernel security mechanisms. Then, we have further developed a new keylogger called DLOGGER, which is more stealthy than existing keyloggers. Instead of injecting any malicious code, it only alters kernel data and leverages existing benign kernel code to build a covert channel, through which attackers can steal sensitive information. Therefore, existing defense mechanisms including those deployed at hypervisor level that search for hidden processes/hidden modules, or monitor kernel code integrity, will not be able to detect DLOGGER. To counter against kernel data attack, by classifying kernel data into different categories and handling them separately, we have proposed an effective defense mechanism and evaluated its efficacy with real experiments. Our experimental results have shown that our defense is effective in detecting kernel data attack with negligible performance overhead.

2 Description of the Specific Problems

When a system is compromised, attackers commonly leave malicious programs behind so as to allow the attackers to: (1) regain the privileged access to the compromised system without re-exploiting a vulnerability, and (2) collect additional sensitive information such as user credentials and financial records. To achieve these two goals, attackers have developed various kernel rootkits. Over the past years, kernel rootkits have posed serious security threats to computing systems. To defend against kernel level malware, a vast variety of approaches have been proposed. These approaches, either rely on additional hardware, or leverage the virtualization technology for countering kernel level attacks. With these defense mechanisms, we can ensure the integrity of kernel code and read-only data, protect kernel hooks from being subverted to compromise kernel control flow, and prevent malicious code from running at the kernel level. Thus, most existing kernel level attacks can be effectively thwarted.

Therefore, attackers are aggressively seeking new vulnerabilities inside the kernel. Ideally, the new attacks should not inject any malicious code running at the kernel level. To this end, kernel data attack has already attracted some attention. By altering kernel data only, without injecting any malicious code, attackers are able to manipulate kernel behaviors. Compared to existing kernel level malware, kernel data attack is more stealthy. This is because, most kernel code does not change during its whole lifetime, and thus, can be well monitored and protected with existing defenses. In contrast, most kernel data is supposed to be inherently changeable (except for read-only data), making it much harder to detect kernel data attacks.

3 Major Research Activities

In this project, we have first assumed the role of attackers and explore the attack space of kernel data attack. Through novel kernel data manipulation, we have demonstrated that kernel data attacks can introduce security threats as serious as existing kernel rootkits, including disabling various kernel-level security mechanisms and stealing sensitive information. And then we have investigated, from the defenders perspective, how to detect kernel data attack. The major research activities of this project are summarized as follows

- We have systematically studied the attack space of kernel data attack. After analyzing Linux kernel source code, we have revealed that the attack space is enormous: in one of the latest Linux Kernel version (3.1.10), there are around 380,000 global function pointers and global variables in the Linux kernel, and the vast majority of these data are subject to change during runtime.

- By examining various Linux kernel internal defense mechanisms, we have observed that the runtime behaviors of these mechanisms rely on some global kernel data. Altering these in-memory global kernel data, attackers can subvert these defense mechanisms. More specifically, we have demonstrated that attackers can tamper with the Linux auditing framework, subvert the Linux AppArmor security module, and bypass NULL pointer dereference mitigation, on a victim machine. Thus, it is clear that kernel data attacks are realistic threats, even as serious as existing kernel rootkits, yet more stealthy than existing kernel rootkits, as they do not require the injection of any kernel-level malicious code.

- To further demonstrate the severity of kernel data attack, we have designed and implemented a novel keylogger: DLOGGER. DLOGGER exploits an inherent property of the Linux proc file system, which is the bridge between the kernel space and the user space. In particular, by redirecting a proc file system pointer to a tty buffer, attackers can construct a covert channel, and then utilize this covert channel to monitor user input and steal sensitive information, such as passwords. DLOGGER is more stealthy than existing keyloggers, as it neither changes any kernel code nor runs a hidden process, which enables it to evade existing rootkit/keylogger detection tools

- We have developed a defense solution to detect kernel data attack. Our defense is built on the fact that there are different types of kernel data, which demonstrate different running behaviors and characteristics during runtime. By providing a kernel data classification and treating different types of data separately, we have shown that the proposed defense is effective in detecting kernel data attack with negligible performance overhead.

4 Key Outcomes

The project started in April, 2015, and has supported two Ph.D. students for their security and system research. As scheduled, we have systematically developed the proposed kernel data attacks and explored the effective defense mechanism, which classifies kernel data into four different types and handles these different types of kernel data separately. We have published two journal papers in IEEE/ACM Transactions on Networking and IEEE Transactions on Information Forensics & Security, as well as ten conference papers in IEEE S&P 2015, WWW 2015, USENIX Security 2015, IEEE DSN 2015, IEEE SRDS 2015, SecureComm 2015, IEEE ICNP 2015, USENIX LISA 2015, IEEE INFOCOM 2016, and IEEE ICAC 2016. We have also filed a U.S. patent titled as "Using Hardware Features for Increased Debugging Transparency".

4.1 Paper Awards

- Best Paper Award, USENIX LISA 2015.
- Best Paper Nominee, WWW 2015.
- Best Paper Nominee, IEEE ICNP 2015.

4.2 Graduated Ph.D. Students

- Jidong Xiao, December 2015.
- Zhang Xu, April 2016.